

AperTO - Archivio Istituzionale Open Access dell'Università di Torino

## Parallel algorithms for nonlinear diffusion by using relaxation approximation

### This is the author's manuscript

*Original Citation:*

*Availability:*

This version is available <http://hdl.handle.net/2318/93506> since 2017-03-13T10:59:43Z

*Publisher:*

Springer-Verlag

*Published version:*

DOI:10.1007/978-3-540-34288-5\_35

*Terms of use:*

Open Access

Anyone can freely access the full text of works made available as "Open Access". Works made available under a Creative Commons license can be used according to the terms and conditions of said license. Use of all other works requires consent of the right holder (author or publisher) if not exempted from copyright protection by the applicable law.

(Article begins on next page)

---

# Parallel algorithms for nonlinear diffusion by using relaxation approximation

Fausto Cavalli<sup>1</sup>, Giovanni Naldi<sup>1</sup> and Matteo Semplice<sup>1</sup>

Department of Mathematics, University of Milano, via Saldini 50 Milano (Italy),  
{cavalli,naldi,semplice}@mat.unimi.it

**Summary.** It has been shown that the equation of diffusion, linear and nonlinear, can be obtained in a suitable scaling limit by a two-velocity model of the Boltzmann equation [6]. Several numerical approximations were introduced in order to discretize the corresponding multiscale hyperbolic systems [8, 1, 4]. In the present work we consider relaxed approximations for multiscale kinetic systems with asymptotic state represented by nonlinear diffusion equations. The schemes are based on a relaxation approximation that permits to reduce the second order diffusion equations to first order semi-linear hyperbolic systems with stiff terms. The numerical passage from the relaxation system to the nonlinear diffusion equation is realized by using semi-implicit time discretization combined with ENO schemes and central differences in space. Finally, parallel algorithms are developed and their performance evaluated. Application to porous media equations in one and two space dimensions are presented.

## 1 Relaxation approximation of nonlinear diffusion

The main aim of this work is to approximate solutions of a nonlinear, degenerate parabolic equation

$$\frac{\partial u}{\partial t} = \Delta(g(u)) \quad (1)$$

for  $x \in \Omega \subset \mathbb{R}^d$ ,  $d \geq 1$ ,  $t \geq 0$ , with suitable boundary conditions and initial condition  $u(x, 0) = u_0(x)$ , where  $g$  is a non-decreasing Lipschitz continuous function on  $\mathbb{R}$ , the degenerate case corresponding to  $g(0) = 0$ . This framework is so general that it includes the porous medium equation and the Stefan problem as well as a wide class of mildly nonlinear parabolic equations. Using the same idea which is at the basis of the well-known relaxation schemes for hyperbolic conservation laws [5], it is possible to develop stable numerical schemes for diffusion and for transport reaction-diffusion equations. In the case of the nonlinear diffusion operator, by introducing an additional variable  $\mathbf{v}(x, t) \in \mathbb{R}^d$  and the positive parameter  $\varepsilon$ , we have the following relaxation system

$$\begin{cases} \frac{\partial u}{\partial t} + \operatorname{div}(\mathbf{v}) = 0 \\ \frac{\partial \mathbf{v}}{\partial t} + \frac{1}{\varepsilon} \nabla g(u) = -\frac{1}{\varepsilon} \mathbf{v} \end{cases} \quad (2)$$

Formally, in the small relaxation limit,  $\varepsilon \rightarrow 0^+$ , the system (2) approximates to leading order equation (1). In order to have a non singular transport operator, by using a suitable parameter  $\Phi$  we can rewrite system (2) as

$$\begin{cases} \frac{\partial u}{\partial t} + \operatorname{div}(\mathbf{v}) = 0 \\ \frac{\partial \mathbf{v}}{\partial t} + \Phi^2 \nabla g(u) = -\frac{1}{\varepsilon} \mathbf{v} + \left(\Phi^2 - \frac{1}{\varepsilon}\right) \nabla g(u) \end{cases} \quad (3)$$

Then, using an auxiliary variable  $w(x, t) \in \mathbb{R}$  we get

$$\begin{cases} \frac{\partial u}{\partial t} + \operatorname{div}(\mathbf{v}) = 0 \\ \frac{\partial \mathbf{v}}{\partial t} + \Phi^2 \nabla w = -\frac{1}{\varepsilon} \mathbf{v} + \left(\Phi^2 - \frac{1}{\varepsilon}\right) \nabla w \\ \frac{\partial w}{\partial t} + \operatorname{div}(\mathbf{v}) = -\frac{1}{\varepsilon} (w - g(u)) \end{cases} \quad (4)$$

In the previous systems the parameter  $\varepsilon$  has physical dimensions of a time and represents the relaxation time, i.e. the characteristic time to reach the equilibrium point in the evolution of the variable  $\mathbf{v}$  governed by the stiff second equation of (4). For consistency,  $w$  has the same dimensions as  $u$ , while each component of  $\mathbf{v}$  has the dimension of  $u$  times a velocity; finally  $\Phi$  is a velocity. Equations (4) form a semilinear hyperbolic system with characteristic velocities  $0, \pm\Phi$ . The parameter  $\Phi$  allows the use of this system with non-stiff velocities (in fact, when  $\Phi = 0$  these velocities are instead  $0, \pm\frac{1}{\varepsilon}$ ).

One of the main advantages of this approach resides in the semilinearity of the system, that is all the nonlinearities are in the (stiff) source terms, while the differential operator is linear. Moreover we point out that degenerate parabolic equations often model physical situations with free boundaries or discontinuities: we expect that schemes for hyperbolic systems will be able to reproduce faithfully these details of the solution. Finally, the relaxation approximation does not exploit the form of the nonlinear function  $g$  and hence it gives rise to a numerical scheme that, to a large extent, is independent of it, resulting in a very versatile tool.

In the following section we will describe the scheme that we used to integrate the relaxed version of (4), i.e. when  $\varepsilon = 0$ , and we will discuss its properties. Numerical results for 1D and 2D cases will be presented in section 3 together with the performance of the parallel implementation.

## 2 The numerical scheme

For simplicity, we will consider a regular rectangular grid on  $\mathbb{R}^d$ . For  $d = 1$  this consists of a set of equally spaced grid points  $x_{i+1/2}$ ,  $i = \dots, -1, 0, 1, \dots$ , with uniform mesh width  $\Delta x = x_{i+1/2} - x_{i-1/2}$ . When  $d > 1$ , we consider the obvious generalization and  $i$  will represent a multi-index of  $d$  integers. The discrete time levels  $t^n$  for  $n = 0, 1, 2, \dots$  ( $t^0 = 0$ ) are also spaced uniformly with time step  $\Delta t$ . As usual we

denote by  $U_i^n$  the approximate value of  $U$  at the centre of the cell  $[x_{i-1/2}, x_{i+1/2}]$  at time  $t^n$ . The spatial discretization of the relaxation system and the corresponding numerical fluxes is realized here by using ENO techniques [10]. This allows us to get highly accurate spatial reconstruction of the solution. The resulting semi-discrete approximation of (4) is of the form

$$\frac{\partial z_i(t)}{\partial t} + F(z_{i-r}(t), \dots, z_{i+s}(t)) = G(z_{i-r}(t), \dots, z_{i+s}(t)) \quad (5)$$

where  $z$  denotes the collection  $(u, v_1, \dots, v_d, w)$  of all the variables appearing in (4) and the stencil used for approximating  $z_i$  is  $r + s + 1$  points wide. In (5),  $F$  is a discretization of the linear differential operator appearing on the left hand side of (4), while  $G$  is a discretization of the non-linear and stiff source terms of (4).

In order to avoid severe restrictions on the time step, we need to couple this high order in space scheme with a time integrator of equal accuracy. Moreover, due to the structure of (5) sketched above, we wish to treat implicitly the time integration of  $G$ , which is stiff, and explicitly the one of  $F$ , which is linear. This is achieved with IMEX schemes tailored to relaxation systems [2, 9].

Numerical tests [5, 8] suggest that the difference between the results obtained with  $\varepsilon \ll 1$  and those with  $\varepsilon = 0$  are negligible. However the relaxed scheme (with  $\varepsilon = 0$ ) gives immediately the projection of the solution onto the equilibrium state in the relaxation step; hence it is simpler to implement. For this reason we consider here only the relaxed scheme.

## 2.1 Implicit relaxed step

The structure of the system to be solved implicitly is of particular importance. By using simply the backward Euler formula, the values  $(u_i^{(1)}, v_i^{(1)}, w_i^{(1)})$  of the solutions of the system  $z_t = G(z)$  with initial data  $(u_i^n, v_i^n, w_i^n)$  may be computed by solving

$$\begin{cases} \frac{u^{(1)} - u^n}{\Delta t} = 0 \\ \frac{v^{(1)} - v^n}{\Delta t} = -\frac{v^{(1)}}{\varepsilon} + \left(\Phi^2 - \frac{1}{\varepsilon}\right) \hat{\nabla} w^{(1)} \\ \frac{w^{(1)} - w^n}{\Delta t} = -\frac{w^{(1)} - g(u^{(1)})}{\varepsilon} \end{cases}$$

where we have suppressed the spatial index  $i$  for clarity and  $\hat{\nabla}$  is a suitable discretization of the spatial gradient. Formally, in the limit  $\varepsilon \rightarrow 0^+$ , this reduces to

$$u^{(1)} = u^n, \quad v^{(1)} = -\hat{\nabla} w^{(1)}, \quad w^{(1)} = g(u^{(1)}). \quad (6)$$

We note here that the first equation is immediately solved and the remaining two decouple. Hence even in the implicit step, we do not need an implicit solver. Up to second order precision in space, the usual 3-point central difference approximation of the derivative is suitable. For degree 3 and 4 one needs however the 5-point approximation or otherwise the quality of the solution and rate of convergence is degraded. Formula (6) represent the relaxation step of our relaxed schemes.

## 2.2 Explicit step

For each explicit step of the IMEX scheme one has to advance from time  $t^n$  to time  $t^{n+1}$  the system  $z_t + F(z) = 0$ , i.e.

$$\frac{\partial}{\partial t} \begin{pmatrix} u \\ v \\ w \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & \Phi^2 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix} = 0 \quad (7)$$

with initial data set to the values  $(u_i^{(1)}, v_i^{(1)}, w_i^{(1)})$  obtained from the relaxation step, as described previously. The characteristic variables  $U, V, W$  move at speed  $\Phi, -\Phi$  and 0 respectively. By changing variables to diagonalize the system we need to reconstruct via ENO only the two fields  $U(x)$  and  $V(x)$  and calculate the corresponding numerical fluxes.

In two space dimensions, the above system generalizes as

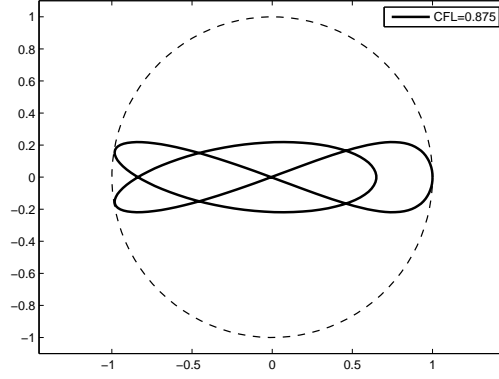
$$\frac{\partial}{\partial t} \begin{pmatrix} u \\ v_{(1)} \\ v_{(2)} \\ w \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \Phi^2 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} u \\ v_{(1)} \\ v_{(2)} \\ w \end{pmatrix} + \frac{\partial}{\partial y} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Phi^2 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} u \\ v_{(1)} \\ v_{(2)} \\ w \end{pmatrix} = 0 \quad (8)$$

where  $\mathbf{v} = (v_{(1)}, v_{(2)})$  and similarly in higher dimensions. We note that only one of the fields  $v_{(i)}$  appear in the differential operator along the  $i^{\text{th}}$  direction. One may then calculate the fluxes separately for each spatial direction by using the aforementioned ENO reconstructions on the fields  $v_{(i)}$  for  $i = 1, \dots, d$ .

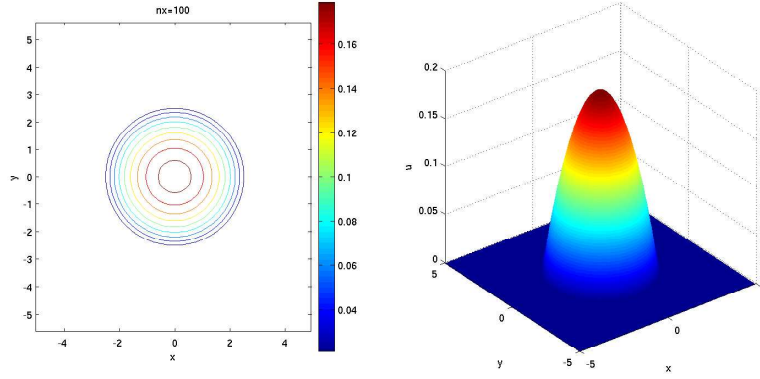
In order to analyse some stability properties of the scheme we consider the linear case  $g(u) = u$ . Moreover we adopt the simplest IMEX scheme represented by the combination of a backward Euler timestep for  $z_t = G(z)$  followed by a forward Euler timestep for  $z_t + F(z) = 0$ . For example when  $F$  is approximated with upwind fluxes and linear reconstructions and the spatial derivative in  $G$  with the central differences formula, this results in a scheme for the variable  $u$  for which a simple Von Neumann analysis reveals the necessity of a CFL condition of the form  $\Delta t \approx C \Delta x^2$  with the numerical estimate  $C \leq 0.875$  for the constant  $C$ . In Figure 1 we show an example of stability regions obtained with the Von Neumann analysis cited above.

## 3 Numerical results

As a numerical test we consider the porous media equation, which corresponds to the choice  $g(u) = u^2$  in (1) and we use relaxed schemes, i.e.  $\varepsilon = 0$ . We adopt periodic boundary conditions but more general boundary conditions may be easily implemented. In order to perform some test for the accuracy of the proposed schemes we consider numerical solutions in comparison with the exact 2D Barenblatt solution. The contour plot of the numerical solution at time  $T = 2$  with  $200 \times 200$  grid points is given in In Figure 3 we show the behaviour of the free boundary, points for the



**Fig. 1.** An example of the stability regions for linear diffusion,  $g(u) = u$ .

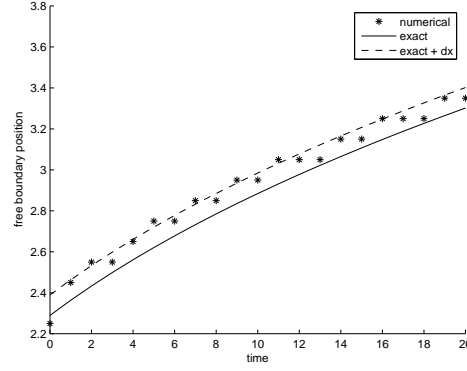


**Fig. 2.** Contour plot of the numerical solution for porous media equation.

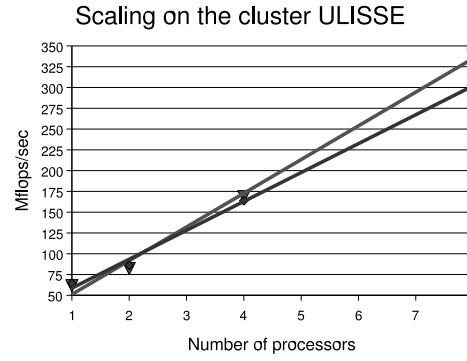
passage from positive to zero value, of the numerical solution and of the Barenblatt solution. In Figure 3 the continuous line represents the exact solution, while the stars are the numerical front. The error between the true front and the approximate front appears of order  $\Delta x$  (the dashed line is at distance  $\Delta x$  from the exact front).

### 3.1 Parallel code

The parallel implementation has been realized through a decomposition of the computational domain by a balanced subdivision of the nodes among the processors set at the beginning of the program. Each processor solves its local problem, using MPI communications to get the boundary data needed. Since both the ENO subroutines and the program to solve (1) with the relaxed scheme do not involve nonlinear solvers of any kind (see Section 2), we expect a linear scaling of the solution time with the number of processors. For portability and easy extension, both the ENO library and the program are written exploiting the PETSC libraries. In Figure 4 we report the



**Fig. 3.** Approximation of the free boundary with 200 grid points along a cross section.

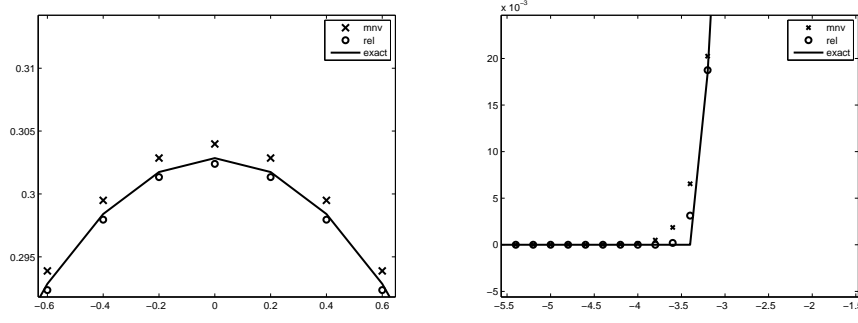


**Fig. 4.** Scaling of the parallel code on a Linux cluster with 72 Intel Xeon processors. When the subproblems assigned to each processor become too small, the time spent exchanging MPI messages among the processors become predominant and the overhead of MPI communications shows up as reduced speedup on the smaller grid.

scaling plot for our parallel code. The algorithm shows a good, almost linear, scaling behaviour, until the subproblems assigned to each processor become too small and MPI communications slow the code down.

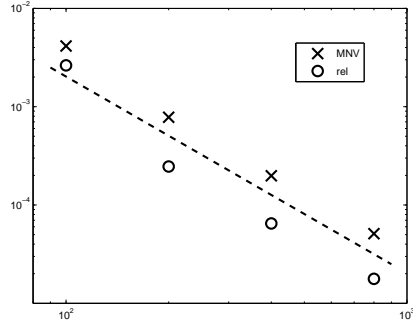
### 3.2 Comparison with another method

We compared our numerical results with those obtained with a linear method proposed and studied, among others, by Berger, Brezis, Rogers [3] and by Magenes, Nochetto and Verdi [7]. Their method is based on the non-linear Chernoff formula and it does not give an explicit formula for the solution, which is instead found by solving a linear problem. It is thus more costly than our method where only matrix-vector products are needed. In the following we refer to this method as the BBRMN method. In Figure 5 we present a comparison of two numerical solutions



**Fig. 5.** Comparison of two numerical solutions of the porous media equation obtained with the relaxation and the BBRMNV method, together with the exact Barenblatt solution. 100 grid points were used. We show particulars of the areas around the maximum point (left hand figure) and around the moving front (right hand figure).

N	BBRMNV	rel
100	4.21e-3	2.75e-3
200	7.73e-4	2.58e-4
400	1.98e-4	6.51e-5
800	5.05e-5	1.83e-5



**Fig. 6.** Comparison of the 1-norm of the error of the relaxation and the BBRMNV method. The exact Barenblatt solution at time  $t = 2$  was used as reference to compute the errors. The dotted line is a reference decay of second order.

of the porous medium equation ( $g(u) = u^2$ ) and the exact self-similar solution due to Barenblatt. The final time of all the simulations is  $T = 2$ . One may see that the higher numerical diffusion of the BBRMNV method shows up both as a lesser accuracy in the neighbourhood of the maximum at  $x = 0$  and as a lower precision in the neighbourhood of the front. The solution represented by dots in Figure 5 was obtained with the relaxation method described in this paper, using spatial ENO reconstructions of degree 2 and an IMEX timestepping procedure with the same accuracy. Both methods under comparison are of second order, as shown by the table and the plot of the 1-norm of the error against the number of grid points that is shown in Figure 6. In any case we point out that the BBRMNV method, however, has been studied more extensively than our technique based on relaxation and adaptive versions should now be implementable. Finally, both methods are easily generalizable to different functions  $g(u)$  and also to equations of the form  $u_t - \Delta(g(u)) = f$ .



## 4 Concluding remarks

In this work we briefly described a class of relaxed schemes for nonlinear and degenerate diffusion problems in any space dimension and in a rectangular domain. Using suitable relaxation approximation we are able to formulate a general diffusion equation into the form of a semilinear system. Then we coupled ENO schemes and IMEX approach for spatial and, respectively, time discretization. We can develop high order methods and we proposed a “blackbox” scheme: it does not exploit the form of nonlinear term  $g(u)$  for the solution. Moreover, in this scheme we avoid the use of nonlinear solvers and, in the relaxed IMEX version, we don’t need solvers at all. The numerical methods can be easily extended from 1D to higher dimensions and parallelized. A comparison with another method is also shown and differences in computational cost and accuracy are outlined. In forthcoming works we will perform theoretical study of the stability of relaxed schemes in the nonlinear case and we will explore the possibility to introduce non structured (rectangular) grids near fronts. Finally, the parallel implementation of WENO reconstructions are under study with interesting preliminary numerical results.

## References

1. Aregba-Driollet D., Natalini R., Tang S.Q.: Diffusive kinetic explicit schemes for nonlinear degenerate parabolic systems. Quaderno IAC N. 26/2000, Roma (2000)
2. Asher U., Ruuth S., Spiteri R.J.: Implicit-explicit Runge-Kutta methods for time dependent Partial Differential Equations. Appl. Numer. Math. **25**, 151–167 (1997)
3. Berger A.E., Brezis H., Rogers J.C.W.: A numerical method for solving the problem  $u_t - \Delta f(u) = 0$ . RAIRO numerical analysis 13, 297–312 (1979)
4. Jin S., Pareschi L., Toscani G.: Diffusive relaxation schemes for multiscale discrete velocity kinetic equations. SIAM J. Numer. Anal., **35**, 2405–2439 (1998)
5. Jin S., Xin, Z.: The relaxation schemes for systems of conservation laws in arbitrary space dimension. Comm. Pure and Appl. Math., **48**, 235–276 (1995)
6. Lions P.L., Toscani G.: Diffusive limit for two-velocity Boltzmann kinetic models. Rev. Mat. Iberoamericana, **13**, 473–513 (1997)
7. Magenes E., Nochetto R.H., Verdi C.: Energy error estimates for a linear scheme to approximate nonlinear parabolic problems. RAIRO Modl. Math. Anal. Numr. 21, 655–678 (1987)
8. Naldi G., Pareschi L.: Numerical schemes for hyperbolic systems of conservation laws with stiff diffusive relaxation. SIAM J. Numer. Anal., **37**, 1246–1270 (2000)
9. Pareschi L., Russo G.: Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation. J. Sci. Comp. (to appear)
10. Shu C.W.: ENO and WENO schemes for hyperbolic conservation laws. In Quarteroni A. (ed) Advanced Numerical Approximation of Nonlinear Hyperbolic Equations. LN in Mathematics, **1697**, Springer, Berlin Heidelberg New York (1998)